CLAIMS

What is claimed is:

1. A method of facilitating data flow between a synchronous process and an asynchronous process, comprising:

converting an input asynchronous data flow from the synchronous process into a synchronous data flow;

processing the synchronous data flow by means of a synchronous task;

converting the processed synchronous data flow into an output asynchronous data flow; and

feeding the output asynchronous data flow to the asynchronous process.

2. The method according to claim 1, further comprising creating a synchronous buffer queue for the input asynchronous data flow.

3. The method according to claim 2, further comprising dequeuing a plurality of input buffers from the synchronous buffer queue.

4. The method according to claim 3, further comprising enqueuing the processed synchronous data flow on an asynchronous buffer queue.

5. The method according to claim 4, wherein processing the synchronous data flow comprises sorting the synchronous data flow.

6. The method according to claim 5, further comprising enqueuing the sorted synchronous data flow to a plurality of output buffers.

7. The method according to claim 6, further comprising synchronously filling the output buffers with the sorted synchronous data flow.

8. The method of claim 6, wherein the number of the output buffers is limited to a predetermined maximum value.

9. The method of claim 6, further comprising saving the sorted synchronous data flow in the output buffers at a record processor until the output buffers are requested by the asynchronous process.

10. The method of claim 1, further comprising saving the processed the synchronous data flow for an image copy restore task.

11. A computer program product having instruction codes for facilitating data flow between a synchronous process and an asynchronous process, the computer program product comprising:
a first set of instruction codes for converting an input asynchronous data flow from the synchronous process into a synchronous data flow;
a second set of instruction codes for processing the synchronous data flow by means of a synchronous task;
a third set of instruction codes for converting the processed synchronous data flow into an output asynchronous data flow; and
a fourth set of instruction codes for feeding the output asynchronous data flow to the asynchronous process.

12. The computer program product according to claim 11, further comprising a fifth set of instruction codes for creating a synchronous buffer queue for the input asynchronous data flow.

13. The computer program product according to claim 12, further comprising a sixth set of instruction codes for dequeuing a plurality of input buffers from the synchronous buffer queue.

14. The computer program product according to claim 13, further comprising a seventh set of instruction codes for enqueuing the processed synchronous data flow on an asynchronous buffer queue.

15. The computer program product according to claim 14, wherein the second set of instruction codes invokes a sort product to synchronously sort the synchronous data flow.

16. The computer program product according to claim 15, further comprising an eighth set of instruction codes for enqueuing the sorted synchronous data flow to a plurality of output buffers.

17. The computer program product according to claim 16, wherein the fourth set of instruction codes fills the output buffers with the sorted synchronous data flow.

18. The computer program product of claim 16, wherein the number of the output buffers is limited to a predetermined maximum value.

19. The computer program product of claim 16, further comprising a ninth set of instruction codes for saving the sorted synchronous data flow in the output buffers at a record processor until the output buffers are requested by the asynchronous process.

20. The computer program product of claim 11, wherein the third set of instruction codes saves the processed the synchronous data flow for an image copy restore task.

21. A system for facilitating data flow between a synchronous process and an asynchronous process, the system comprising:

an asynchronous to synchronous server for converting an input asynchronous data flow from the synchronous process into a synchronous data flow;

a synchronous task for processing the synchronous data flow; and

a synchronous to asynchronous server for converting the processed synchronous data flow into an output asynchronous data flow, and for feeding the output asynchronous data flow to the asynchronous process.

22. The system according to claim 21, further comprising a synchronous buffer queue for storing the input asynchronous data flow.

23. The system according to claim 22, further comprising a task for dequeuing a plurality of input buffers from the synchronous buffer queue.

24. The system according to claim 23, further comprising a task for enqueuing the processed synchronous data flow on an asynchronous buffer queue.

25. The system according to claim 24, wherein the synchronous task comprises a sorter that synchronously sorts the synchronous data flow

26. The system according to claim 25, further comprising a task for enqueuing the sorted synchronous data flow to a plurality of output buffers.

27. The system according to claim 26, wherein the synchronous to asynchronous server fills the output buffers with the sorted synchronous data flow.

28. The system of claim 26, wherein the number of the output buffers is limited to a predetermined maximum value.

29. The system of claim 26, wherein the synchronous to asynchronous server saves the sorted synchronous data flow in the output buffers at a record processor until the output buffers are requested by the asynchronous process.

30. The system of claim 21, wherein the synchronous to asynchronous server saves the processed the synchronous data flow for an image copy restore task.